Quadrature Rules on Triangles and Tetrahedra for Multidimensional Summation-By-Parts Operators

Zelalem Arega Worku[†] · **Jason E. Hicken**[‡] · **David W. Zingg**[†]

Abstract Multidimensional diagonal-norm summation-by-parts (SBP) operators with collocated volume and facet nodes, known as diagonal-E operators, are attractive for entropy-stable discretizations from an efficiency standpoint. However, there is a limited number of such operators, and those currently in existence often have a relatively high node count for a given polynomial order due to a scarcity of suitable quadrature rules. We present several new symmetric positive-weight quadrature rules on triangles and tetrahedra that are suitable for construction of diagonal-E SBP operators. For triangles, quadrature rules of degree one through twenty with facet nodes that correspond to the Legendre-Gauss-Lobatto (LGL) and Legendre-Gauss (LG) quadrature rules are derived. For tetrahedra, quadrature rules of degree one through ten are presented along with the corresponding facet quadrature rules. All of the quadrature rules are provided in a supplementary data repository. The quadrature rules are used to construct novel SBP diagonal-E operators, whose accuracy and maximum time-step restrictions are studied numerically.

Keywords Quadrature · Cubature · Summation-by-parts · Multidimensional · Triangle · Tetrahedron

Mathematics Subject Classification (2020) 65M06 · 65M12 · 65N06 · 65N12

1 Introduction

Summation-by-parts (SBP) operators enable the construction of entropy-stable high-order discretizations of the Euler and Navier-Stokes equations [9,4,1,10,24,5,27]. Diagonal-norm SBP operators have collocated solution and volume quadrature nodes, which enables straightforward inversion of the norm/mass matrix, facilitating efficient implementation of high-order methods with explicit time-marching schemes. Due to the collocation, the efficiency of the method is significantly affected by the number of quadrature nodes. This is even more prominent for the Hadamard-form entropy-stable discretizations [9,4] of the Euler and Navier-Stokes equations on simplices, as a volume flux computation coupling each node with all other nodes must be calculated. Therefore, the development of quadrature rules with fewer nodes on simplices is imperative to improve efficiency of entropy-stable discretizations with diagonal-norm SBP operators. While existing positive interior (PI) quadrature rules offer the fewest nodes for a given quadrature accuracy, their use for entropy-stable SBP discretizations requires expensive element coupling computations, although there are ways to reduce this cost to some extent [1,28]. Alternatives to PI rules for SBP operators have been proposed; notably, Hicken et al. [13] derived quadrature rules with a set of volume nodes on each facet of the triangle and tetrahedron, and Chen and Shu [2] presented rules on the triangle that enable construction of SBP operators with collocated volume and facet quadrature nodes, which are referred to as diagonal-E or R^0[20] SBP operators. Diagonal-E SBP operators eliminate the need to extrapolate the solution from volume to facet nodes and are of particular interest for entropy-stable SBP discretizations as they reduce the cost of element coupling operations and enable straightforward enforcement of boundary conditions. However, the number of nodes required for the quadrature rules of existing diagonal-E operators is significantly larger than that of the PI rules, especially for the tetrahedron. Furthermore, only a limited number of rules of this type are available in the literature. In light of this, the goal of this paper is to find efficient quadrature rules on the triangle and tetrahedron for the purpose of constructing efficient diagonal-norm diagonal-E multidimensional SBP operators.

Quadrature rules with boundary nodes on simplices have been explored, although to a lesser extent than PI rules. Sets of nodes on the triangle that are typically well-suited for interpolation or quadrature accuracy have both been utilized to derive such rules, e.g., see [29,3,23,11,31,33,19] among others. Similar

Zelalem Arega Worku
E-mail: zelalem.worku@mail.utoronto.ca

Jason E. Hicken
E-mail: hickej2@rpi.edu

David W. Zingg
E-mail: dwz@oddjob.utias.utoronto.ca

[†]Institute for Aerospace Studies, University of Toronto, Toronto, Ontario, M3H 5T6, Canada [‡]Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY, United States

studies for the tetrahedron, however, are lacking. A shared property of several of the nodal sets in the mentioned studies is that for a degree p operator there are $p+1$ nodes on each facet of the triangle, and vertices are included. This automatically excludes the rules from being applicable for construction of diagonal-E SBP operators, as the facet quadrature accuracy is not sufficient. We recall that a sufficient condition on the facet quadrature rule for the existence of a degree p SBP operator is that it be at least of degree $2p$ accurate. Although this is only a sufficient condition, as is evident from the existence of Legendre-Gauss-Lobatto (LGL) tensor-product operators on quadrilaterals and hexahedra, to the authors' knowledge, no diagonal-E SBP operator on simplices with facet quadrature rule of degree less than $2p$ has been constructed.

Odd-degree quadrature rules on the triangle for diagonal-E SBP operators with Legendre-Gauss (LG) facet nodes were first introduced in [2], followed by even degree rules with LGL and LG facet nodes in [5, 12] and odd-degree rules with LGL facet nodes in [36]. Together, these studies provide quadrature rules up to degree eight[1] on the triangle. For the tetrahedron, Hicken [12] derived even degree quadrature rules up to degree eight with PI rule facet nodes. Marchildon and Zingg [20] studied optimization of these operators and developed novel rules up to degree four on the tetrahedron, managing to lower the number of nodes for the degree two and four quadrature rules. This was achieved by allowing the facet nodes to be placed at the vertices and edges of the tetrahedron. Other efforts to improve the efficiency of entropy-stable discretizations with multidimensional SBP operators include, for instance, the use of staggered grids in [5], entropy-split formulations in [35], and collapsed coordinate tensor-product elements in [22].

In this paper, we derive symmetric quadrature rules for construction of diagonal-norm diagonal-E SBP operators on triangles and tetrahedra using the open-source Julia code `SummationByParts.jl` (v0.2.0) [12], which employs the Levenberg-Marquardt algorithm (LMA) [18,21] to solve the nonlinear systems of equations that arise from the quadrature accuracy conditions. The code's capability is enhanced by enforcing a constraint to find positive weights and by combining it with a particle swarm optimization (PSO) [16] subroutine to mitigate issues related to initial guesses and convergence to suboptimal local minima. We extend the available set of quadrature rules for diagonal-E operators up to degree twenty on triangles with both the LGL and LG facet node configurations, and up to degree ten for tetrahedra, achieving a significant reduction in the number of nodes relative to many of the existing rules. The new rules are used to construct novel diagonal-E SBP operators, whose accuracy and time-step restrictions are studied numerically.

The rest of the paper is organized as follows: Section 2 describes the problem statement and symmetry groups on the reference triangle and tetrahedron, Section 3 details the methodology employed, Section 4 presents the derived quadrature rules along with a description of multidimensional SBP operators and their construction, and numerical results are presented in Section 5 followed by conclusions in Section 6.

2 Preliminaries

Constructing quadrature rules over a domain of interest requires solving highly nonlinear systems of equations to find the nodal locations and weights. Usually, quadrature rules are designed to be exact for a desired degree of polynomial functions. The problem can be stated as: find $\boldsymbol{x}$ and $\boldsymbol{w}$ such that

$$\int_{\Omega} \mathcal{P}_j\left(\hat{\boldsymbol{x}}\right) \mathrm{d}\Omega = \sum_{i=1}^{n_p} w_i \mathcal{P}_j\left(\boldsymbol{x}_i\right), \qquad j \in \{1, \ldots, n_b\}, \tag{2.1}$$

where $\hat{\boldsymbol{x}} = [x_1, \ldots, x_d]^T$, d is the spatial dimension, $\boldsymbol{x}$ is the vector of the coordinate tuples of the nodes, $\boldsymbol{w}$ is the vector of quadrature weights, and n_p and n_b denote the number of quadrature points and polynomial basis functions, respectively. For a degree q_v accurate quadrature rule on a simplex, there are $n_b = \binom{p+d}{d}$ polynomial basis functions. Rewriting the problem statement in matrix form, we have

$$\boldsymbol{g} \coloneqq \mathsf{V}^T \boldsymbol{w} - \boldsymbol{f} = \boldsymbol{0}, \tag{2.2}$$

where V is the Vandermonde matrix containing evaluations of the basis functions at each node along its columns and $\boldsymbol{f} = \left[\int_{\Omega} \mathcal{P}_1\left(\widehat{\boldsymbol{x}}\right) \mathrm{d}\Omega, \ldots, \int_{\Omega} \mathcal{P}_{n_b}\left(\widehat{\boldsymbol{x}}\right) \mathrm{d}\Omega\right]^T$. The basis functions used to construct the Vandermonde matrix affect its condition number. It is well-known that, for high-order polynomials, monomial basis functions result in an ill-conditioned V. In contrast, the orthonormal Proriol-Koornwinder-Dubiner (PKD) [26,17,7] basis functions offer better conditioning and a convenient $\boldsymbol{f}$ vector; all except the first entry of $\boldsymbol{f}$ are zero due to the orthogonality of the basis functions.

For the purpose of constructing a degree p diagonal-norm diagonal-E multidimensional SBP operator, we require that:

[1] [12] (v0.1.0) provides additional rules of degree 12 and 16 with LGL facet nodes.

i. all quadrature points lie in the closure of the simplex,
ii. all weights are positive,
iii. the volume quadrature is at least degree $q_v = 2p - 1$ accurate,
iv. a subset of the quadrature points lying on each facet form a positive-weight facet quadrature rule of at least degree $q_f = 2p$, and
v. both the facet and volume quadrature rules are symmetric.

The symmetry requirement ensures that a solution obtained using the SBP operators is not spatially biased within an element. Furthermore, the symmetry constraints reduce the number of unknowns in (2.2) substantially [32].

The reference triangle and tetrahedron are defined, respectively, as

$$\Omega_{\text{tri}} = \{(x, y) \mid x, y \geq -1;\ x + y \leq 0\}, \tag{2.3}$$
$$\Omega_{\text{tet}} = \{(x, y, z) \mid x, y, z \geq -1;\ x + y + z \leq -1\}. \tag{2.4}$$

There are three symmetry groups on the triangle, and five on the tetrahedron [8,38,32]. However, we will identify symmetric nodes that lie on the facets of the simplices as being in separate symmetry groups. On the triangle, the symmetry groups, in barycentric coordinates, are permutations of

$$\begin{aligned} S_1 &= \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right), \quad S_{21} = (\alpha, \alpha, 1 - 2\alpha), \qquad S_{111} = (\alpha, \beta, 1 - \alpha - \beta), \\ S_{\text{vert}} &= (1, 0, 0), \qquad S_{\text{mid-edge}} = \left(\frac{1}{2}, \frac{1}{2}, 0\right), \quad S_{\text{edge}} = (\alpha, 1 - \alpha, 0), \end{aligned} \tag{2.5}$$

where α and β are parameters such that the quadrature points lie in the closure of the domain. The symmetry groups in the first line of (2.5) represent interior points, while those in the second line denote points on the facets. Similarly, the symmetry groups on the reference tetrahedron are permutations of

$$\begin{aligned} S_1 &= \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right), & S_{\text{face-cent}} &= \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0\right), \\ S_{31} &= (\alpha, \alpha, \alpha, 1 - 3\alpha), & S_{\text{vert}} &= (1, 0, 0, 0), \\ S_{22} &= (\alpha, \alpha, 1 - \alpha, 1 - \alpha), & S_{\text{mid-edge}} &= \left(\frac{1}{2}, \frac{1}{2}, 0, 0\right), \\ S_{211} &= (\alpha, \alpha, \beta, 1 - 2\alpha - \beta), & S_{\text{face-21}} &= (\alpha, \alpha, 1 - 2\alpha, 0), \\ S_{1111} &= (\alpha, \beta, \gamma, 1 - \alpha - \beta - \gamma), & S_{\text{edge}} &= (\beta, 1 - \beta, 0, 0), \\ & & S_{\text{face-111}} &= (\alpha, \beta, 1 - \alpha - \beta, 0). \end{aligned} \tag{2.6}$$

The facet symmetry groups in the right column of (2.6) are special cases of the symmetry groups in the left column, which are used in this work to denote interior points exclusively.

3 Methodology

The Vandermonde matrix in (2.2) is a function of the quadrature points, $\boldsymbol{x}$; hence, the algorithm to solve the equation starts by guessing the nodal locations and weights. This is done indirectly by providing the type and number of symmetry groups and the values of the associated parameters and weights. Using the initial guess of the parameters, it is possible to compute the coordinates of the i-th node using the transformation,

$$\boldsymbol{x}_i = \mathsf{T}^T \boldsymbol{\lambda}_k, \tag{3.1}$$

where $\mathsf{T} \in \mathbb{R}^{(d+1)\times d}$ contains the coordinates of the $d+1$ vertices in its rows and $\boldsymbol{\lambda}_k$ is the k-th permutation of the barycentric coordinates of the symmetry group that corresponds to the i-th node. The weight vector, $\boldsymbol{w}$, is constructed by assigning equal weights to all nodes in the same symmetry group.

To derive a degree q_v quadrature rule satisfying all the properties required to construct a degree p SBP diagonal-E operator, we first need to find a facet quadrature rule of degree $q_f \geq 2p$. On the reference triangle, we use either the LGL rule with $n_f = p + 2$ facet nodes (including the vertices) or the LG rule with $n_f = p + 1$. In construction of the volume quadrature rule, we fix the facet quadrature points; hence, the parameters in the facet symmetry groups are kept constant, i.e., we solve for the weights at all points and for the parameters associated with the interior symmetry groups. A similar strategy is followed to find the quadrature rules on the tetrahedron. While existing PI rules can be used as facet quadrature rules for the tetrahedron, they generally lead to more volume nodes than necessary. Hence, we first construct facet

Table 3.1 Number of tetrahedron volume nodes due to inclusion of facet symmetry groups

group	$S_{\text{face-cent}}$	S_{vert}	$S_{\text{mid-edge}}$	$S_{\text{face-21}}$	S_{edge}	$S_{\text{face-111}}$
# nodes	4	4	6	12	12	24

quadrature rules of degree $2p$ that would result in fewer volume quadrature points on the tetrahedron by placing some of the nodes at the vertices and/or edges of the facets. We note that, depending on their nodal locations, symmetry groups with the same number of nodes on the triangle can produce a different number of nodes on the tetrahedron when applied to its facets. For example, each of the S_{vert} and $S_{\text{face-21}}$ symmetry groups results in three nodes per facet, but four and twelve volume nodes, respectively. The number of volume nodes due to inclusion of the various facet symmetry groups of the tetrahedron is presented in Table 3.1.

The problem in (2.2) can be written equivalently as a minimization problem,

$$\min_{\boldsymbol{\tau}} \frac{1}{2}\boldsymbol{g}^T\boldsymbol{g}, \tag{3.2}$$

where $\boldsymbol{\tau} = [\widehat{\boldsymbol{\lambda}}, \widehat{\boldsymbol{w}}]^T$, and $\widehat{\boldsymbol{\lambda}}$ and $\widehat{\boldsymbol{w}}$ are vectors of all the parameters and weights associated with each symmetry group. The LMA is widely used to solve (3.2) as it is less sensitive to initial guesses than Newton's method. The LMA computes the step direction, $\boldsymbol{h}$, which is initialized as the zero vector, as

$$\widetilde{\boldsymbol{h}} = -\widetilde{\mathsf{A}}^{+}\widetilde{\mathsf{J}^T\boldsymbol{g}}, \tag{3.3}$$

where $\mathsf{A} = \mathsf{J}^T\mathsf{J} + \nu\text{diag}(\mathsf{J}^T\mathsf{J})$, and $\nu > 0$ controls the scale of exploration. The initialization of $\boldsymbol{h}$ as the zero vector is important, as the parameters that correspond to the facet node locations must remain unchanged; updating the parameter vector as in (3.6) below will update only the entries of the parameter vector corresponding to the nonzero entries of $\boldsymbol{h}$. The symbol $\widetilde{(\cdot)}$ denotes extraction of the rows and columns of a vector or matrix that correspond to the parameters of the interior symmetry groups and all weights, which are updated iteratively. Note that the notation $(\cdot)^+$ in (3.3) denotes the Moore-Penrose pseudo-inverse, $\mathsf{J} \in \mathbb{R}^{n_b \times n_\tau}$ is the Jacobian matrix given by

$$\mathsf{J}_{(i,j)} = \frac{\partial \boldsymbol{g}_i}{\partial \boldsymbol{\tau}_j}, \tag{3.4}$$

and n_τ is the sum of the number of parameters and weights. The Jacobian can also be written in terms of block matrices as

$$\mathsf{J} = \left[\sum_{k=1}^{d} \mathsf{V}_{x_k}^T \text{diag}(\boldsymbol{w}) \frac{\partial \boldsymbol{x}_{(:,k)}}{\partial \widehat{\boldsymbol{\lambda}}}, \mathsf{V}^T \frac{\partial \boldsymbol{w}}{\partial \widehat{\boldsymbol{w}}}\right], \tag{3.5}$$

where V_{x_k} is the k-th direction derivative of V and $\boldsymbol{x}_{(:,k)}$is the k-th direction component vector of $\boldsymbol{x}$. The matrix $\partial \boldsymbol{x}_{(:,k)}/\partial \widehat{\boldsymbol{\lambda}}$ is computed using the relation in (3.1), and $\partial \boldsymbol{w}/\partial \widehat{\boldsymbol{w}}$ is a matrix of zeros and ones. The value of ν is initially set to 1000, but it is gradually reduced or increased depending on the convergence of the objective function.

The algorithm starts with an initial guess, $\boldsymbol{\tau}^{(0)}$, and the value of $\boldsymbol{\tau}$ at the n-th iteration is updated as

$$\boldsymbol{\tau}^{(n+1)} = \boldsymbol{\tau}^{(n)} + \eta^{(n)}\boldsymbol{h}^{(n)}, \tag{3.6}$$

where $\eta^{(n)} = 1$ is used unless a negative weight is encountered. If a negative weight is encountered at the i-th entry of $\boldsymbol{\tau}^{(n+1)}$, then the update is recomputed using

$$\eta^{(n)} = \frac{(\varepsilon - \boldsymbol{\tau}_i^{(n)})}{\boldsymbol{h}_i^{(n)}}, \tag{3.7}$$

where $\varepsilon > 0$ is an arbitrary lower bound for the update of the negative weight, and is set to $\varepsilon = 10^{-4}$ in all of our cases.

Despite being more robust than Newton's method, the LMA still suffers from poor initial guesses; especially, as the number of parameters grows and the quadrature accuracy increases, often stagnating at a suboptimal local minimum. To mitigate these issues, the LMA is coupled with a particle swarm optimization (PSO) algorithm. The PSO algorithm starts with an initialization of n_c particles, each with a random initial guess of $\boldsymbol{\tau}$. The objective function, $\boldsymbol{g}$, given in (2.2), is computed for each particle and the

personal best, $\boldsymbol{\tau}_{pb}$, and global best, $\boldsymbol{\tau}_{gb}$, approximations are tracked throughout the iterations. The PSO step size or velocity vector, $\boldsymbol{v}$, which is initialized as the zero vector, is computed for each particle as

$$\widetilde{\boldsymbol{v}}^{(n+1)} = b\widetilde{\boldsymbol{v}}^{(n)} + c_1\boldsymbol{r}_1 \circ \left(\widetilde{\boldsymbol{\tau}}_{pb}^{(n)} - \widetilde{\boldsymbol{\tau}}^{(n)}\right) + c_2\boldsymbol{r}_2 \circ \left(\widetilde{\boldsymbol{\tau}}_{gb}^{(n)} - \widetilde{\boldsymbol{\tau}}^{(n)}\right), \tag{3.8}$$

where $b = 0.6$ is the inertial weight parameter, $c_1 = 1.5$ is the cognitive parameter, $c_2 = 1.5$ is the social parameter, $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are vectors of length equal to that of $\widetilde{\tau}$ with uniform random entries on $[0, 1]$, and $\circ$ denotes an elementwise multiplication. The vector, $\boldsymbol{\tau}$, is updated at each iteration as

$$\boldsymbol{\tau}^{(n+1)} = \boldsymbol{\tau}^{(n)} + \boldsymbol{v}^{(n+1)}. \tag{3.9}$$

If a negative weight is encountered for any particle, it is simply replaced by a small positive number, e.g., 10^{-4}, and the update is recomputed. As the quadrature accuracy is increased, the algorithm sometimes stagnates at a local minimum and further exploration is hindered. If the same local minimum is obtained over a number of iterations, then $\widetilde{\boldsymbol{\tau}}$ is perturbed as $(1-\delta)\widetilde{\boldsymbol{\tau}} + \delta\boldsymbol{r}$, where $\boldsymbol{r}$ is a vector of length equal to the length of $\widetilde{\boldsymbol{\tau}}$ with uniform random entries on $[0, 1]$ and $\delta > 0$ is an arbitrary small number. The choice of the value of δ depends on the quadrature degree, as the sensitivity to perturbation increases with the quadrature degree.

The PSO and LMA are coupled in such a way that the output vector of one is used as an initial vector of the other in a loop until convergence to machine precision. The PSO mitigates issues related to initial guesses and convergence to suboptimal local minima, while the LMA offers fast convergence when good initial values are provided. Despite the efficiency of the coupled algorithm, at very high quadrature degrees, the minimization sometimes stagnates before convergence is realized. In such cases, the minimization is restarted, and in some instances, the parameters associated with the interior nodes are initialized using parameters of known PI rules.

4 Quadrature rules and SBP operators

An SBP operator on a compact reference domain, $\hat{\Omega}$, with a piecewise smooth boundary, $\hat{\Gamma}$, is defined as [13]:

Definition 4.1 $\mathsf{D}_{\hat{\boldsymbol{x}}_i} \in \mathbb{R}^{n_p \times n_p}$ is a degree p SBP operator in the i-direction approximating the first derivative $\frac{\partial}{\partial \hat{\boldsymbol{x}}_i}$ on the set of nodes $S = \{\boldsymbol{x}_j\}_{j=1}^{n_p}$ if

1. $[\mathsf{D}_{\hat{\boldsymbol{x}}_i}\boldsymbol{p}]_j = \frac{\partial \mathcal{P}}{\partial \hat{\boldsymbol{x}}_i}(\boldsymbol{x}_j)$ for all $\mathcal{P} \in \mathbb{P}^p(\hat{\Omega})$
2. $\mathsf{D}_{\hat{\boldsymbol{x}}_i} = \mathsf{H}^{-1}\mathsf{Q}_{\hat{\boldsymbol{x}}_i}$, where H is a symmetric positive definite matrix, and
3. $\mathsf{Q}_{\hat{\boldsymbol{x}}_i} + \mathsf{Q}_{\hat{\boldsymbol{x}}_i}^T = \mathsf{E}_{\hat{\boldsymbol{x}}_i}$, where

$$\boldsymbol{p}^T\mathsf{E}_{\hat{\boldsymbol{x}}_i}\boldsymbol{q} = \int_{\hat{\Gamma}} \mathcal{P}\mathcal{Q}\, \boldsymbol{n}_{\hat{\boldsymbol{x}}_i}\, \mathrm{d}\Gamma\,, \quad \forall \mathcal{P}, \mathcal{Q} \in \mathbb{P}^r(\hat{\Gamma}),$$

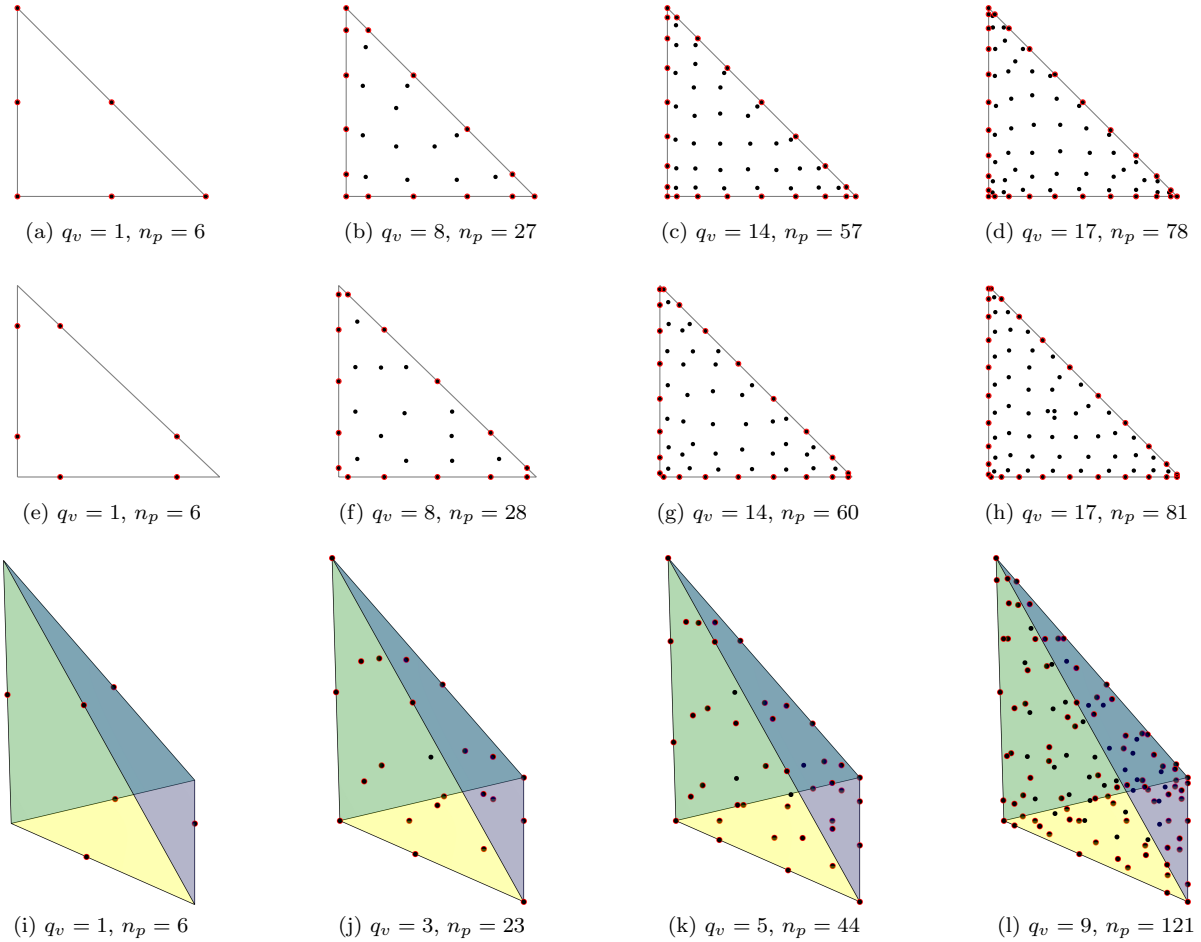
where $r \geq p$, $\boldsymbol{n}_{\hat{\boldsymbol{x}}_i}$ is the i-component of the outward pointing unit normal vector on $\hat{\Gamma}$, and $\mathbb{P}^q$ denotes a polynomial space of degree q.

A diagonal-norm SBP operator has a diagonal H matrix containing the weights of a volume quadrature rule of degree at least $q_v = 2p - 1$ [13,14]. The existence of a sufficiently accurate positive-weight quadrature rule on $\hat{\Omega}$ is necessary and sufficient for the existence of a degree p first-derivative diagonal-norm SBP operator [13]. The boundary operator, $\mathsf{E}_{\hat{\boldsymbol{x}}_i}$, is also constructed using a degree $2p$ accurate positive-weight facet quadrature rule as [6]

$$\mathsf{E}_{\hat{\boldsymbol{x}}_i} = \sum_{\gamma \in \hat{\Gamma}} \mathsf{R}_\gamma^T \mathsf{B}_\gamma \mathsf{N}_{\hat{\boldsymbol{x}}_i\gamma} \mathsf{R}_\gamma, \tag{4.1}$$

where $\mathsf{N}_{\hat{\boldsymbol{x}}_i\gamma}$ is a diagonal matrix containing the i-component of the outward unit normal vector on facet γ, R_γ is an extrapolation operator from the volume to the facet nodes, and B_γ is a diagonal matrix containing the facet quadrature weights. If the volume and facet quadrature nodes are collocated, then R_γ simply picks out function values at the facet nodes, resulting in a diagonal $\mathsf{E}_{\hat{\boldsymbol{x}}_i}$ matrix. The collocation of the facet and volume nodes reduces the cost of element coupling via simultaneous approximation terms (SATs), especially for entropy-stable discretizations. For further discussion on construction of multidimensional SBP operators, we refer the reader to [13,6]. Construction of SATs for SBP discretizations of various model equations in CFD can be found in [6,4,28,37,34,35].

Using the methodology outlined in the previous section, we have derived quadrature rules that satisfy conditions i. – v. stated in Section 2, and constructed SBP diagonal-E operators on the reference triangle and tetrahedron. Quadrature rules with LGL and LG facet nodes of degree up to twenty on the triangle and



(a) $q_v = 1$, $n_p = 6$ (b) $q_v = 8$, $n_p = 27$ (c) $q_v = 14$, $n_p = 57$ (d) $q_v = 17$, $n_p = 78$

(e) $q_v = 1$, $n_p = 6$ (f) $q_v = 8$, $n_p = 28$ (g) $q_v = 14$, $n_p = 60$ (h) $q_v = 17$, $n_p = 81$

(i) $q_v = 1$, $n_p = 6$ (j) $q_v = 3$, $n_p = 23$ (k) $q_v = 5$, $n_p = 44$ (l) $q_v = 9$, $n_p = 121$

Fig. 4.1 Nodal locations of selected quadrature rules with LGL facet nodes (top row), LG facet nodes (middle row), and rules on the tetrahedron (bottom row). The symbols • and ○ denote the volume and facet nodes, respectively, which are collocated on the facets.

up to ten on the tetrahedron are derived, which can be found in the supplementary data repository[2]. Fig. 4.1 illustrates the nodal configurations of some of the quadrature rules in 2D and 3D. Many of the rules are novel to the authors' knowledge, and substantial improvements, in terms of number of quadrature points, have been achieved relative to several of the existing rules on the tetrahedron, as illustrated in Table 4.1. These improvements result in more efficient SBP diagonal-E operators; for instance, new operators of degree 3 and 4 are constructed with 44 and 76 nodes, respectively, instead of 69 and 99 nodes. The derived quadrature rules extend the available set of SBP diagonal-E operators from degree 4 to 10 in 2D and from degree 4 to 5 in 3D. Unless specified otherwise, all the numerical studies in this work use SBP diagonal-E operators with quadrature rules stated in the first columns of the different types of quadrature rules presented in Table 4.1, which also provides the minimum nodal spacing of the quadrature rules on the reference elements. It is noted that all rules on the triangle with LGL facet nodes have larger minimum nodal spacing than those with LG facet nodes. Furthermore, the rules obtained on the tetrahedron have equal or larger minimum nodal spacing than existing rules.

5 Numerical results

In this section, the SBP diagonal-E operators constructed using the proposed quadrature rules are applied to linear and nonlinear problems. First, a mesh is generated by partitioning the spatial domain, Ω, into m squares or cubes in each direction and subdividing them into two triangles or six tetrahedra, respectively. The nodes on the physical elements are obtained by affinely mapping the nodes on the reference elements. The 2D and 3D meshes are refined for mesh convergence studies using $m_k = 60 - 5p + (12 - p)k$ and $m_k = 10 + 5k$ number of edges in each direction, respectively, where $k = \{1, 2, \dots\}$ denotes the refinement

[2] https://github.com/OptimalDesignLab/SummationByParts.jl/tree/zw_v0.2.0/quadrature_data

Table 4.1 Number of quadrature points, n_p, and minimum quadrature node spacing, Δr, on the reference triangle and tetrahedron. Quadrature rules in the cited columns denote existing results, while novel quadrature rules are underlined.

q_v	Tri-LGL				Tri-LG				Tet					
	n_p	Δr	n_p[5,12]	n_p[36]	n_p	Δr	n_p[12]	n_p[2]	n_p	Δr	n_p[12]	Δr[12]	n_p[20]	Δr[20]
1	6	1.000		6	6	0.423		6	6	1.000			6	1.000
2	7	0.471	7		7	0.423	7		7	0.866	13	0.333	7	0.866
3	10	0.553		10	10	0.225		10					23	0.222
4	12	0.325	12		12	0.225	12		23†	0.377	36	0.183	26	0.222
5	15	0.345		15	18	0.083		18	44	0.237				
6	18	0.214	18		21	0.097	21		51	0.159	69	0.106		
7	24	0.141		24	22	0.094		22	76	0.107				
8	27	0.129	27		28	0.094	28		89	0.094	99	0.017		
9	33	0.129			34	0.068			121	0.077				
10	36	0.076			39	0.068			145‡	0.070				
11	40	0.128			42	0.051								
12	48	0.069	48		49	0.051								
13	55	0.039			54	0.024								
14	57	0.062			60	0.040								
15	69	0.074			69	0.032								
16	72	0.052	75		72	0.032								
17	78	0.030			81	0.026								
18	93	0.041			93	0.026								
19	96	0.037			96	0.010								
20	103	0.037			103	0.022								

† The $q_v = 3$ and $q_v = 4$ quadrature rules on the tetrahedron are identical.
‡ A rule with 139 nodes is provided in the supplementary data repository, but it is not considered here since it leads to a restrictively small time step.

level. The number of edges are chosen to ensure that errors are sufficiently larger than machine precision, enabling calculations of convergence rates for the highest-degree operators.

The standard fourth-order Runge-Kutta (RK4) scheme is applied to march the numerical solution in time. For the accuracy studies, sufficiently small time steps are used such that the temporal errors are negligible compared to the spatial errors. As in [28,35], the L^2 solution error in the domain is computed by interpolating the numerical solution from the SBP nodes to a quadrature rule of degree $3p+1$, integrating the square of the solution error, summing the result over all elements, and taking the square root of the sum.

5.1 Linear advection problem

We consider the linear advection equation,

$$\frac{\partial \mathcal{U}}{\partial t} + \sum_{i=1}^{d} \boldsymbol{c}_i \frac{\partial \mathcal{U}}{\partial \hat{\boldsymbol{x}}_i} = 0, \tag{5.1}$$

on the periodic domain $\Omega = [0,1]^d$. The problem is used to test the accuracy and time step stability limits of the operators. The initial condition, $\mathcal{U}(\hat{\boldsymbol{x}}, t=0)$, and the exact solution, $\mathcal{U}(\hat{\boldsymbol{x}}, t)$, for the problem are given by

$$\mathcal{U}(\hat{\boldsymbol{x}}, t) = \prod_{i=1}^{d} \sin(\omega\pi(\hat{\boldsymbol{x}}_i - \boldsymbol{c}_i t)), \tag{5.2}$$

where $\boldsymbol{c} = [5/4, \sqrt{7}/4]^T$ in 2D or $\boldsymbol{c} = [3/2, 1/2, 1/\sqrt{2}]^T$ in 3D is used in all cases. The values of $\boldsymbol{c}$ are chosen to set the wave speed magnitude at $\sqrt{d}$ but are otherwise chosen randomly. The direction of the wave propagation depends on $\boldsymbol{c}$ and affects numerical errors and mesh convergence rates in some cases.

The advection equation, (5.1), is discretized using the diagonal-E SBP operators and an upwind SAT, see, e.g., [6] for the details of the SBP-SAT discretization. The problem is run up to $t = 1$ with the ω parameters in (5.2) set to 8 and 2 for the 2D and 3D cases, respectively. The solution errors and convergence rates are tabulated in Table 5.1, which shows convergence rates close to $p + 1$ on the finest meshes.

Table 5.1 Solution convergence of the linear advection problem discretized with SBP diagonal-E operators on triangles and tetrahedra.

p	q_v		Tri-LGL			Tri-LG			Tet		
			m_1	m_2	m_3	m_1	m_2	m_3	m_1	m_2	m_3
1	1	error	3.81e-01	2.98e-01	2.37e-01	1.69e-01	1.28e-01	9.99e-02	7.74e-02	4.72e-02	3.13e-02
		rate	–	1.60	1.72	–	1.80	1.86	–	1.72	1.84
1	2	error	3.58e-02	2.36e-02	1.64e-02	4.15e-02	2.77e-02	1.94e-02	8.80e-03	4.42e-0	2.64e-03
		rate	–	2.69	2.72	–	2.62	2.66	–	2.39	2.30
2	3	error	8.60e-04	4.75e-04	2.87e-04	1.23e-03	6.76e-04	4.04e-04			
		rate	–	3.85	3.76	–	3.90	3.85			
2	4	error	5.27e-04	3.19e-04	2.09e-04	5.56e-04	3.37e-04	2.20e-04	9.11e-04	3.56e-04	1.75e-04
		rate	–	3.26	3.17	–	3.26	3.17	–	3.26	3.19
3	5	error	4.11e-05	2.22e-05	1.31e-05	5.24e-05	2.81e-05	1.63e-05	5.92e-05	1.94e-05	8.01e-06
		rate	–	3.99	3.98	–	4.05	4.07	–	3.89	3.96
3	6	error	4.18e-05	2.26e-05	1.33e-05	4.52e-05	2.44e-05	1.43e-05	4.48e-05	1.45e-05	5.98e-06
		rate	–	3.99	3.99	–	4.00	4.00	–	3.93	3.95
4	7	error	4.72e-06	2.21e-06	1.15e-06	4.02e-06	1.87e-06	9.68e-07	3.44e-06	8.43e-07	2.85e-07
		rate	–	4.91	4.93	–	4.94	4.95	–	4.89	4.86
4	8	error	3.63e-06	1.70e-06	8.82e-07	4.35e-06	2.08e-06	1.09e-06	2.70e-06	6.28e-07	2.08e-07
		rate	–	4.91	4.93	–	4.79	4.85	–	5.07	4.95
5	9	error	3.51e-07	1.41e-07	6.30e-08	4.31e-07	1.72e-07	7.81e-08	1.44e-07	2.64e-08	7.14e-09
		rate	–	5.94	6.01	–	5.95	5.92	–	5.90	5.86
5	10	error	3.65e-07	1.49e-07	6.88e-08	3.54e-07	1.43e-07	6.51e-08	9.62e-08	1.78e-08	4.80e-09
		rate	–	5.82	5.78	–	5.90	5.89	–	5.87	5.86
6	11	error	4.27e-08	1.44e-08	5.52e-09	4.60e-08	1.54e-08	5.95e-09			
		rate	–	7.07	7.16	–	7.08	7.14			
6	12	error	4.68e-08	1.58e-08	6.16e-09	4.59e-08	1.57e-08	6.21e-09			
		rate	–	7.05	7.04	–	6.96	6.94			
7	13	error	1.26e-08	3.62e-09	1.23e-09	1.34e-08	3.83e-09	1.30e-09			
		rate	–	8.08	8.07	–	8.13	8.11			
7	14	error	1.25e-08	3.61e-09	1.23e-09	1.29e-08	3.77e-09	1.30e-09			
		rate	–	8.06	8.05	–	8.00	7.99			
8	15	error	4.46e-09	1.16e-09	3.57e-10	4.55e-09	1.20e-09	3.74e-10			
		rate	–	8.72	8.85	–	8.65	8.72			
8	16	error	4.93e-09	1.30e-09	4.04e-10	4.80e-09	1.24e-09	3.80e-10			
		rate	–	8.65	8.75	–	8.77	8.87			
9	17	error	5.48e-09	1.36e-09	3.63e-10	4.91e-09	1.17e-09	3.22e-10			
		rate	–	9.06	9.87	–	9.28	9.69			
9	18	error	4.78e-09	1.14e-09	3.07e-10	5.02e-09	1.21e-09	3.32e-10			
		rate	–	9.32	9.79	–	9.22	9.70			
10	19	error	2.59e-08	5.03e-09	1.19e-09	2.73e-08	5.29e-09	1.25e-09			
		rate	–	10.63	10.77	–	10.64	10.82			
10	20	error	2.63e-08	5.17e-09	1.24e-09	2.62e-08	5.10e-09	1.21e-09			
		rate	–	10.55	10.71	–	10.62	10.76			

In addition to the accuracy of the operators, we are also interested in their maximum time-step limits for explicit time marching schemes. A large time-step limit is desired for stability-bounded problems, where the maximum stable time step can be applied without compromising accuracy. The maximum time step is computed for each operator using golden section optimization. For this study, the triangular and tetrahedral meshes are obtained by subdividing quadrilateral and hexahedral meshes with four elements in each direction and ω is set to 2 for both the 2D and 3D cases. The discretization is considered to be stable if the change in energy is less than or equal to zero after five periods. The change in energy at a given time step is computed as,

$$\Delta E = \sum_{\Omega_k \in \mathcal{T}_h} \left(\boldsymbol{u}_k^T \mathsf{H}_k \boldsymbol{u}_k - \boldsymbol{u}_{0,k}^T \mathsf{H}_k \boldsymbol{u}_{0,k} \right), \tag{5.3}$$

Table 5.2 Maximum time step on the reference triangle and tetrahedron.

q_v	Tri-LGL	Tri-LG	Tet	Tet[12]	Tet[20]
1	0.0258	0.0758	0.0641		0.0641
2	0.0446	0.0394	0.0388	0.0345	0.0388
3	0.0323	0.0365			0.0164
4	0.0235	0.0216	0.0065	0.0208	0.0204
5	0.0258	0.0203	0.0184		
6	0.0146	0.0131	0.0097	0.0101	
7	0.0145	0.0129	0.0074		
8	0.0106	0.0092	0.0049	0.0058	
9	0.0070	0.0089	0.0075		
10	0.0066	0.0012	0.0059		
11	0.0008	0.0066			
12	0.0059	0.0015			
13	0.0016	0.0040			
14	0.0048	0.0040			
15	0.0025	0.0035			
16	0.0038	0.0020			
17	0.0022	0.0027			
18	0.0033	0.0024			
19	0.0029	0.0023			
20	0.0025	0.0019			

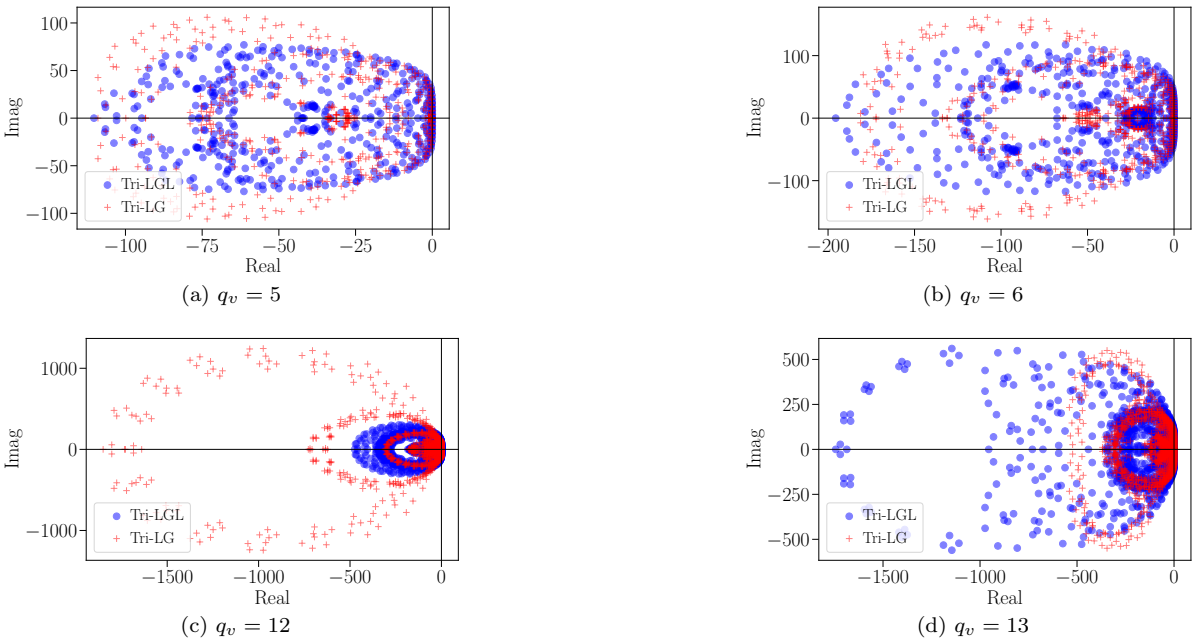
Table 5.3 Reciprocal of the spectral radii of the discretization system matrices for the 2D advection problem.

q_v	1	2	3	4	5	6	7	8	9	10
Tri-LGL	0.0090	0.0150	0.0111	0.0081	0.0091	0.0051	0.0051	0.0037	0.0025	0.0023
Tri-LG	0.0278	0.0143	0.0132	0.0079	0.0075	0.0049	0.0048	0.0034	0.0033	0.0004
q_v	11	12	13	14	15	16	17	18	19	20
Tri-LGL	0.0003	0.0021	0.0006	0.0017	0.0009	0.0013	0.0008	0.0012	0.0011	0.0009
Tri-LG	0.0025	0.0005	0.0015	0.0015	0.0013	0.0007	0.0010	0.0009	0.0009	0.0007

where $\boldsymbol{u}_k$, $\boldsymbol{u}_{0,k}$, and H_k are the solution vector at the specified time step, the initial solution vector, and the norm matrix on element Ω_k, respectively, and $\mathcal{T}_h$ is a set containing all physical elements.

Table 5.2 presents the maximum time-step values for each SBP-SAT discretization. On the triangle, we have not made improvements in terms of number of nodes relative to the existing quadrature rules except in the case of the degree 16 rule with the LGL facet nodes; hence, comparisons of the maximum time steps with previously existing operators are not presented. On the tetrahedron, the new quadrature rule for the degree 2 SBP diagonal-E operator yields a smaller stable time step than the existing rules, while the degree 3 and 4 operators with $q_v = 2p$ quadrature rules lead to slightly lower but comparable stable time steps relative to the existing rules. The table also shows that the degree 3 and 4 diagonal-E operators constructed with $q_v = 2p - 1$ quadrature rules have about 1.72 and 1.28 times larger maximum time steps, respectively, than the existing degree 3 and 4 operators with $q_v = 2p$. This, combined with their lower node count, leads to substantial efficiency improvements for stability-bounded problems.

A similar conclusion on the relative time-step restrictions can be obtained using the eigen-spectra of the system matrix arising from the spatial discretization. Table 5.3 presents the reciprocal of the spectral radii of the system matrices for the two-dimensional periodic advection problem discretized using SBP diagonal-E triangular elements with LGL and LG facet nodes. The mesh is identical to that used for calculating the maximum stable time steps in Table 5.2. As expected, for each pair of degree q_v triangular operators with LGL and LG facet nodes, the operator with the larger stable time-step limit exhibits a smaller spectral radius. The ratios of the maximum stable time steps for operators with LGL facet nodes to those with LG face nodes are consistently close to the ratios of the reciprocal of their corresponding spectral radii; hence, the maximum stable time-step and spectral analysis lead to the same conclusion. Some examples of the eigen-spectra of the discretization system matrix obtained using upwind SATs are shown in Fig. 5.1. As illustrated, all of the eigenvalues lie on the left-hand of the imaginary line, producing stable discretizations. We have also verified that the symmetric SATs produce purely imaginary eigenvalues.



(a) $q_v = 5$ (b) $q_v = 6$

(c) $q_v = 12$ (d) $q_v = 13$

Fig. 5.1 Examples of eigen-spectra for discretization of the advection problem using SBP diagonal-E operators and upwind SATs.

5.2 Isentropic vortex problem

The isentropic vortex problem, governed by the Euler equations, is another common test case used to study the accuracy of high-order methods. We consider the 3D case on the periodic domain $\Omega = [-10, 10]^3$ with the initial conditions [30]

$$\begin{aligned}
\rho &= \left(1 - \frac{2}{25}(\gamma - 1)\exp\left(1 - (x_2 - t)^2 - x_1^2\right)\right)^{\frac{1}{\gamma - 1}}, \\
e &= \frac{\rho^\gamma}{\gamma(\gamma - 1)} + \frac{\rho}{2}\left(u^2 + v^2 + w^2\right), \\
u &= -\frac{2}{5}(x_2 - t)\exp\left(\frac{1}{2}\left[1 - (x_2 - t)^2 - x_1^2\right]\right), \\
v &= 1 + \frac{2}{5}x_1\exp\left(\frac{1}{2}\left[1 - (x_2 - t)^2 - x_1^2\right]\right), \\
w &= 0,
\end{aligned} \tag{5.4}$$

where ρ is the density, e is the total energy per unit volume, u, v, and w are the velocities in the x_1, x_2, and x_3 directions, respectively, and $\gamma = 7/5$ is the ratio of specific heats.

We use the Hadamard-form entropy-stable discretization [4] on tetrahedral elements with the Ismail-Roe two-point fluxes [15]. Furthermore, the matrix-type interface dissipation operator of [15] is applied. The problem is run until $t = 1$, and a mesh convergence study is conducted. The L^2 solution errors and their rates of convergence are shown in Table 5.4. Convergence rates greater than $p + 0.5$ are attained on the finest meshes for all operators constructed using the new quadrature rules. This confirms that the derived quadrature rules and SBP diagonal-E operators are suitable for development of efficient high-order SBP-SAT discretizations on simplices.

6 Conclusions

Several novel quadrature rules that are applicable for construction of diagonal-norm diagonal-E SBP operators on triangles and tetrahedra are derived. The quadrature rules are obtained by coupling the LMA and PSO methods to solve the nonlinear systems of equations arising from the quadrature accuracy conditions. The LMA provides fast convergence when a good initial condition is provided, while the PSO enables efficient exploration of the design space while also mitigating stagnation issues at suboptimal local

Table 5.4 Solution convergence of the 3D isentropic vortex problem discretized with SBP diagonal-E operators on tetrahedra.

p	q_v		m_1	m_2	m_3	m_4	m_5
1	1	error	1.53e+00	9.93e-01	7.03e-01	5.27e-01	4.14e-01
		rate	–	1.51	1.55	1.57	1.58
1	2	error	1.35e+00	7.98e-01	5.65e-01	4.12e-01	3.16e-01
		rate	–	1.82	1.55	1.73	1.72
2	4	error	3.73e-01	1.93e-01	1.06e-01	6.68e-02	4.52e-02
		rate	–	2.30	2.68	2.53	2.54
3	5	error	9.12e-02	3.59e-02	1.68e-02	8.87e-03	5.03e-03
		rate	–	3.24	3.40	3.51	3.68
3	6	error	9.27e-02	3.68e-02	1.72e-02	8.89e-03	5.02e-03
		rate	–	3.21	3.40	3.63	3.71
4	7	error	2.72e-02	7.87e-03	2.70e-03	1.22e-03	5.80e-04
		rate	–	4.31	4.80	4.36	4.81
4	8	error	2.82e-02	8.25e-03	2.87e-03	1.26e-03	6.02e-04
		rate	–	4.27	4.72	4.53	4.78
5	9	error	6.03e-03	1.35e-03	4.27e-04	1.51e-04	6.30e-05
		rate	–	5.20	5.17	5.71	5.65
5	10	error	6.47e-03	1.47e-03	4.57e-04	1.60e-4	6.63e-05
		rate	–	5.15	5.25	5.77	5.69

minima. The combination of the PSO and LMA algorithms was crucial for generating new quadrature rules that were previously unattainable with the LMA alone. Furthermore, heuristic consideration of symmetry orbits to optimize node placement was essential for extending results to higher polynomial degrees. Previous techniques did not achieve these results due to the difficulty in finding good initial guesses and the heuristic nature of the choice of nodal symmetry group combinations.

Quadrature rules of degrees one through twenty on triangles with both the LGL and LG type facet nodes are presented. For tetrahedra, quadrature rules of degree one through ten are reported, which, in most cases, have substantially fewer nodes than previously known rules for SBP diagonal-E operators. The newly derived quadrature rules lead to SBP diagonal-E operators with enhanced efficiency relative to many of the existing operators of the same degree. They also extend the available set of SBP diagonal-E operators from degree 4 to 10 in 2D and from degree 4 to 5 in 3D.

The diagonal-norm diagonal-E multidimensional SBP operators are applied to solve the linear advection and isentropic vortex problems on periodic domains. Mesh refinement studies for the problems show that convergence rates on the finest meshes are greater than $p + 0.5$ in all cases. We have found that the quadrature points with the LGL facet nodes on triangles have larger minimum nodal spacing than those with the LG facet nodes. For tetrahedra, the rules constructed in this work provide equal or larger minimum nodal spacing than those found in the literature. We have also investigated the maximum time-step values for stability, and found that most of the new rules offer comparable or larger stable time steps relative to previously reported rules.

Declarations

Conflicts of Interest

The authors have no conflicts of interest to declare.

Data Availability

All the quadrature rules reported in this work can be found in the supplementary data repository at https://github.com/OptimalDesignLab/SummationByParts.jl/tree/zw_v0.2.0/quadrature_data. The software implementation used to obtain the quadrature rules is publicly accessible at https://github.com/OptimalDesignLab/SummationByParts.jl. The package contains examples demonstrating how to extract the quadrature rules and operators derived in this paper, as well as how to use the package to derive new quadrature rules.

Acknowledgments

The authors would like to thank Professor Masayuki Yano and his Aerospace Computational Engineering Lab at the University of Toronto for the use of their software, the Automated PDE Solver (APS). The first author would also like to thank André Marchildon for the discussions on placement of quadrature points on tetrahedral elements. Computation were performed on the Niagara supercomputer at the SciNet HPC Consortium [25]. SciNet is funded by: the Canada Foundation for Innovation; the Government of Ontario; Ontario Research Fund - Research Excellence; and the University of Toronto.

References

1. Chan, J.: On discretely entropy conservative and entropy stable discontinuous Galerkin methods. Journal of Computational Physics **362**, 346–374 (2018)
2. Chen, T., Shu, C.W.: Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws. Journal of Computational Physics **345**, 427–461 (2017)
3. Cohen, G., Joly, P., Roberts, J.E., Tordjman, N.: Higher order triangular finite elements with mass lumping for the wave equation. SIAM Journal on Numerical Analysis **38**(6), 2047–2078 (2001)
4. Crean, J., Hicken, J.E., Del Rey Fernández, D.C., Zingg, D.W., Carpenter, M.H.: Entropy-stable summation-by-parts discretization of the Euler equations on general curved elements. Journal of Computational Physics **356**, 410–438 (2018)
5. Del Rey Fernández, D.C., Crean, J., Carpenter, M.H., Hicken, J.E.: Staggered-grid entropy-stable multidimensional summation-by-parts discretizations on curvilinear coordinates. Journal of Computational Physics **392**, 161–186 (2019)
6. Del Rey Fernández, D.C., Hicken, J.E., Zingg, D.W.: Simultaneous approximation terms for multi-dimensional summation-by-parts operators. Journal of Scientific Computing **75**(1), 83–110 (2018)
7. Dubiner, M.: Spectral methods on triangles and other domains. Journal of Scientific Computing **6**, 345–390 (1991)
8. Felippa, C.A.: A compendium of FEM integration formulas for symbolic work. Engineering Computations **21**(8), 867–890 (2004)
9. Fisher, T.C., Carpenter, M.H.: High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains. Journal of Computational Physics **252**, 518–557 (2013)
10. Gassner, G.J., Winters, A.R., Hindenlang, F.J., Kopriva, D.A.: The BR1 scheme is stable for the compressible Navier–Stokes equations. Journal of Scientific Computing **77**(1), 154–200 (2018)
11. Giraldo, F.X., Taylor, M.A.: A diagonal-mass-matrix triangular-spectral-element method based on cubature points. Journal of Engineering Mathematics **56**, 307–322 (2006)
12. Hicken, J.E.: SummationByParts.jl. URL https://github.com/OptimalDesignLab/SummationByParts.jl
13. Hicken, J.E., Del Rey Fernández, D.C., Zingg, D.W.: Multidimensional summation-by-parts operators: General theory and application to simplex elements. SIAM Journal on Scientific Computing **38**(4), A1935–A1958 (2016)
14. Hicken, J.E., Zingg, D.W.: Summation-by-parts operators and high-order quadrature. Journal of Computational and Applied Mathematics **237**(1), 111–125 (2013)
15. Ismail, F., Roe, P.L.: Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks. Journal of Computational Physics **228**(15), 5410–5436 (2009)
16. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
17. Koornwinder, T.: Two-variable analogues of the classical orthogonal polynomials. In: R.A. Askey (ed.) Theory and Application of Special Functions, pp. 435–495. Academic Press (1975)
18. Levenberg, K.: A method for the solution of certain non-linear problems in least squares. Quarterly of Applied Mathematics **2**(2), 164–168 (1944)
19. Liu, Y., Teng, J., Xu, T., Badal, J.: Higher-order triangular spectral element method with optimized cubature points for seismic wavefield modeling. Journal of Computational Physics **336**, 458–480 (2017)
20. Marchildon, A.L., Zingg, D.W.: Optimization of multidimensional diagonal-norm summation-by-parts operators on simplices. Journal of Computational Physics **411**, 109380 (2020)
21. Marquardt, D.W.: An algorithm for least-squares estimation of nonlinear parameters. Journal of the Society for Industrial and Applied Mathematics **11**(2), 431–441 (1963)
22. Montoya, T., Zingg, D.W.: Efficient tensor-product spectral-element operators with the summation-by-parts property on curved triangles and tetrahedra. Journal of Scientific Computing (Accepted), ArXiv Preprint arXiv:2306.05975 (2023)
23. Mulder, W.A.: Higher-order mass-lumped finite elements for the wave equation. Journal of Computational Acoustics **9**(02), 671–680 (2001)
24. Parsani, M., Carpenter, M.H., Nielsen, E.J.: Entropy stable discontinuous interfaces coupling for the three-dimensional compressible Navier-Stokes equations. J. Comput. Physics **290**, 132–138 (2015)
25. Ponce, M., van Zon, R., Northrup, S., Gruner, D., Chen, J., Ertinaz, F., Fedoseev, A., Groer, L., Mao, F., Mundim, B.C., et al.: Deploying a top-100 supercomputer for large parallel workloads: The Niagara supercomputer. In: Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning), pp. 1–8 (2019)
26. Proriol, J.: Sur une famille de polynomes à deux variables orthogonaux dans un triangle. Comptes Rendus Hebdomadaires des Séances de l'Académie des Sciences **245**(26), 2459–2461 (1957)
27. Ranocha, H.: Comparison of some entropy conservative numerical fluxes for the Euler equations. Journal of Scientific Computing **76**(1), 216–242 (2018)
28. Shadpey, S., Zingg, D.W.: Entropy-stable multidimensional summation-by-parts discretizations on hp-adaptive curvilinear grids for hyperbolic conservation laws. Journal of Scientific Computing **82**(3), 70 (2020)
29. Taylor, M.A., Wingate, B.A., Vincent, R.E.: An algorithm for computing Fekete points in the triangle. SIAM Journal on Numerical Analysis **38**(5), 1707–1720 (2000)

30. Williams, D.M., Jameson, A.: Nodal points and the nonlinear stability of high-order methods for unsteady flow problems on tetrahedral meshes. In: 21st AIAA Computational Fluid Dynamics Conference, AIAA 2013-2830 (2013)
31. Wingate, B.A., Taylor, M.A.: Performance of numerically computed quadrature points. Applied Numerical Mathematics **58**(7), 1030–1041 (2008)
32. Witherden, F., Vincent, P.: On the identification of symmetric quadrature rules for finite element methods. Computers & Mathematics with Applications **69**(10), 1232–1241 (2015)
33. Witherden, F.D., Vincent, P.E.: An analysis of solution point coordinates for flux reconstruction schemes on triangular elements. Journal of Scientific Computing **61**, 398–423 (2014)
34. Worku, Z.A., Zingg, D.W.: Simultaneous approximation terms and functional accuracy for diffusion problems discretized with multidimensional summation-by-parts operators. Journal of Computational Physics **445**, 110634 (2021)
35. Worku, Z.A., Zingg, D.W.: Entropy-split multidimensional summation-by-parts discretization of the Euler and compressible Navier-Stokes equations. Journal of Computational Physics **502**, 112821 (2024)
36. Wu, X., Kubatko, E.J., Chan, J.: High-order entropy stable discontinuous Galerkin methods for the shallow water equations: curved triangular meshes and GPU acceleration. Computers & Mathematics with Applications **82**, 179–199 (2021)
37. Yan, J., Crean, J., Hicken, J.E.: Interior penalties for summation-by-parts discretizations of linear second-order differential equations. Journal of Scientific Computing **75**(3), 1385–1414 (2018)
38. Zhang, L., Cui, T., Liu, H.: A set of symmetric quadrature rules on triangles and tetrahedra. Journal of Computational Mathematics pp. 89–96 (2009)